MULTIMEDIA UNIVERSITY

STUDENT ID NO

MULTIMEDIA UNIVERSITY

FINAL EXAMINATION

TRIMESTER 3, 2015/2016

DCS5038 – PROGRAM DESIGN

(DIT Groups)

30 MAY 2016
2.30 p.m. – 4.30 p.m.
(2 Hours)

INSTRUCTIONS TO STUDENT

1. This question paper consists of **TWELVE (12)** pages. There are **THREE (3)** sections in this booklet.

2. **SECTION A**: There are **FIFTEEN (15)** True and False questions (15 marks). Answer **ALL** questions in the answer booklet provided.

3. **SECTION B:** There are **THREE (3)** structured questions (60 marks). Answer **ALL** questions in the answer booklet provided.

4. **SECTION C**: There are **TWO (2)** questions in this section. Answer **ONLY ONE (1)** question in the answer booklet provided.

SECTION A (15 Marks)

Instruction: *Answer TRUE or FALSE for the questions below and write your answers in the answer booklet provided.*

1. *Pseudocode* is an artificial and informal language that helps programmers to develop algorithms.

2. Lengthy *comments* can cause poor execution-time performance.

3. *Variables* may be defined anywhere inside the body of the main function.

4. The *switch case* selection statement performs an action if a condition is true and performs a different action if the condition is false.

5. The *float* data type should **NOT** be used to control the repetition of a loop.

6. The following is the correct expression to increment *c* by 1.

   c++=1;

7. In a *switch case* statement, a *default case* is mandatory.

8. The following array definition would cause a syntax error.

   int num[5] = {22, 29, 36, 48, 55, 67};

9. Functions are invoked by a *function definition.*

10. The following *function prototype* defines a function called *calcSqrt* which takes a float value as an argument and returns an integer value.

    float calcSqrt (int x);

11. A *function call* must specify the name and arguments of the function.

12. Pointers are variables that contain *memory addresses* as their values.

13. Function *strcmp* returns 1 if its first argument is equal to its second argument.

14. The *dot operator (.)* is used to specify the structure numbers.

15. If an existing file is opened for *writing*, the newly written data is appended to the end of the file.

Continued.....

SECTION B (60 Marks)

Instruction: *Answers all questions in the answer booklet provided.*

QUESTION 1 (20 Marks)

The following description explains how a customer is charged for the rental of a car in a car rental company.

- Initialize all the required variables.
- Get the inputs of *customer's name*, *customer's ID* and *car code.*

Car Model	Car Code	Rent (RM) / day
Honda City	M	250.00
Toyota Vios	X	235.00
Proton Preve	Z	200.00

- Use a *while loop* and *switch case* statement to determine the daily car rental based on the input of *car code* as specified in the table above (user must input only the specified code, otherwise she/he must reenter).
- Get input of the rental *duration.*
- Calculate the *total price* using the formula:

  Totalprice= rental x duration
- Display the *customer's name* and *total price.*

a. Based on the description given above, draw the *flowchart.* [9 marks]
b. Based on the description given above, write the *pseudocode.* [6 marks]
c. Trace and write the output for the following program. [5 marks]

```
#include <stdio.h>

int main()
{
   int x = 3, y = 7, z = 16;
   int a, b, c;

   a = (++x / 2) + (--z / 5);
   b = 4 * x - y++ + a++;
   c = x + y - z;

   printf("a = %d\n", a);
   printf("b = %d\n", b);
   printf("c = %d\n", c);
}
```

Continued.....

QUESTION 2 (20 Marks)

a. There are **SIX (6)** errors in the following program. Write the line numbers that has errors and corrected statement. [3 marks]

```
Line 1   #include<studio.h>
Line 2   int main[]
Line 3   {
Line 4       int num = 1, result, multiplication;
Line 5       printf("\nEnter multiplication table number : ");
Line 6       scanf("&d", %multiplication);
Line 7       do
Line 8       {
Line 9           result = num x multiplication;
Line 10          printf("\n%d x %d = %d", num, multiplication, result);
Line 11          num++;
Line 12      }
Line 13      while(num <= 12)
Line 14  }
```

b. Convert the following mathematical formulas to the proper C expressions. Use appropriate built-in functions from the *math.h* header file. [3 marks]

i. $p + q\left(1 + \frac{1}{r}\right)^3$ ii. $\sqrt{\frac{5AB}{X}}$

c. Write the equivalent *if-else* statement for the following *switch case* statement. [9 marks]

```
switch(code)
{
    case '1': case 'S' : strcpy(day, "Sunday"); break;
    case '2': strcpy(day, "Monday"); break;
    case '3': strcpy(day, "Tuesday"); break;
    case '4': strcpy(day, "Wednesday"); break;
    case '5': strcpy(day, "Thursday"); break;
    case '6': strcpy(day, "Friday"); break;
    case '7': case 'T' : strcpy(day, "Saturday"); break;
    default : printf("You have chosen an invalid code!\n");
    exit(1);
}
printf("Your have chosen %s.\n", day);
```

Continued.....

d. Write a function called *calMovie(...)* that takes in *code* which is of type *char* and *ticket* which is of type *integer* as parameters. Use *if...else* statement to identify the price for each code and finally calculate the total payment. Returns the total payment which is of type *float*. Sample output is given as a guide. The formula for payment is given as;

$$payment = price\ x\ ticket$$

[5 marks]

OUTPUT SCREEN

```
Code        Movies                  Price
*****************************************
C           Civil War              (RM14.00)
X           X-Men Apocalypse       (RM12.00)
T           Tarzan                 (RM10.000)
*****************************************

Enter movie code         : X
Enter number of tickets : 1
Payment: RM 12.00
```

QUESTION 3 (20 Marks)

a. Write the output after the code segment shown below is executed successfully. The statements are related to each other. [5 marks]

```
void main()
{
    int a=5, b=7, c=-9, *x, *y, *z;

     x=&a;
     y=&b;
     z=&c;

     printf("\n x: %d", *x +1);
     printf("\n y: %d", *y *3);
     printf("\n z: %d", *z -7);

     *x=11;
     b=4*a;

     printf("\n a: %d", a);
     printf("\n b: %d\n", b);
}
```

Continued.....

b. Answer the following questions based on the following table:

No.	Account Type	Account Number	Amount
1.	*Saving Account*	*17551003*	*550*
2.	*Current Account*	*50013589*	*2590*
3.	*Credit Card*	*93596540*	*993*
4.	*Hire Purchase*	*33598015*	*33250*

i. Create a structure called *Financial* with the attributes shown above. Set the account type length to 20 characters and account number to 8 characters.

[2 mark]

ii. Create a structure variable called *account* inside the *main()* function with the array size of four.

[1 mark]

iii. Using a *for...loop* statement, prompt the user to enter the *account type*, *account number* and the *amount* and store all values in the structure.

[5 marks]

OUTPUT SCREEN

```
Enter the Account Type   : Saving Account
Enter the Account Number : 17551003
Enter the Account Amount : RM550

Enter the Account Type   : Current Account
Enter the Account Number : 50013589
Enter the Account Amount : RM2590

Enter the Account Type   : Credit Card
Enter the Account Number : 93596540
Enter the Account Amount : RM993

Enter the Account Type   : Hire Purchase
Enter the Account Number : 33598015
Enter the Account Amount : RM33250
```

Continued.....

c. Write the partial codes to satisfy the description given below.

i. Declare a file pointer called *fptr*. Using the file pointer, open a file called *Laptop.txt* for *reading*.

[2 marks]

ii. Check to see the availability of the file. If it is unavailable, display the message "*File could not be opened. Program aborted.*"

[2 marks]

iii. There are two different prices in the file for each laptop's model. Use a *while loop* to extract all the prices from the file to be displayed on the screen.

[3 marks]

Content of Laptop.txt

```
1099 1450
2950 3205
4550 5210
```

OUTPUT SCREEN

```
Discounted Price : 1099.00
Normal Price     : 1450.00

Discounted Price : 2950.00
Normal Price     : 3205.00

Discounted Price : 4550.00
Normal Price     : 5210.00

Discounted Price : 4550.00
Normal Price     : 5210.00
```

Continued.....

SECTION C (25 Marks)

Instruction: *Choose and answer* ***ONLY ONE (1)*** *question from this section and write your answers in the answer booklet provided.*

QUESTION 1 (25 Marks)

You are required to write a **complete program** that calculates the total fees of first year CDP students.

Guide.

- Create a structure called *Record*. The variables in the structure are:
  - *name* : string size 30
  - *course* : string size 50
  - *tuitionFee* : float
  - *discount* : float
  - *totalFee* : float
  - *studentID* : int
  - *courseCode* : int
  - *spm* : int

- In the *main()* function:
  - Create a structure variable array called *student* which has 3 elements.
  - Write the codes to display the menu.
  - Call function *get_info(...)*, passing array *student* as parameter.
  - Call function *get_fees(...)*, passing array *student* as parameter.
  - Call function *display(...)*, passing array *student* as parameter.

- In the *get_info(...)* function:
  - Use *for...loop* (repeat for 3 times):
    - Request from the user to enter data for name, student ID, course code and number of A in SPM.
    - Using *if-else* statement, determine the course and tuition fee based on the table below.

Course code	Course	Tuition fee (RM)
1	Diploma in IT	12,000
2	Diploma in Business Administration	11,250
3	Diploma in Accounting	18,975

Continued.....

- In the *get_fees(...)* function:
  - Use *for...loop (*repeat for 3 times*)*:
    - Determine the discount that the student can get based on the number of As obtained in SPM by using *if..else* statement.

Number of As	Tuition fee discount
9 or 10	50% of tuition fee
7 or 8	25% of tuition fee
5 or 6	15% of tuition fee
4 or below	0% of tuition fee

    - Determine the total fees based on the formula;

$$total\ fee = tuition\ fee - discount$$

- In the *display(...)* function:
  - Use *for...loop (*repeat for 3 times*)*:
    - Display the name, student ID, course, tuition fee, discount and total fee.
- You are required to write the function prototypes for each function used in the program. Use the output scree below as guide.

OUTPUT SCREEN

```
------------------------------
      DIPLOMA COURSES
------------------------------

 1. Diploma in IT
 2. Diploma in Business Administration
 3. Diploma in Accounting

Student 1
Enter your name                          : Ananda
Enter your student ID                    : 1106157
Enter your course                        : 3
Enter the number of A's obtained in SPM : 9

Student 2
Enter your name                          : Amin
Enter your student ID                    : 1106295
Enter your course                        : 1
Enter the number of A's obtained in SPM : 7
```

Continued.....

```
Student 3
Enter your name                              : Andy
Enter your student ID                        : 1106374
Enter your course                            : 2
Enter the number of A's obtained in SPM : 4

--------------------------------
      TUITION FEES DETAILS
--------------------------------

Student 1
Name          : Ananda
Student ID    : 1106157
Course        : Diploma in Accounting
Tuition fees : RM 18975.00
Discount      : RM 9487.50
Total fees    : RM 9487.50

Student 2
Name          : Amin
Student ID    : 1106295
Course        : Diploma in IT
Tuition fees : RM 12000.00
Discount      : RM 3000.00
Total fees    : RM 9000.00

Student 3
Name          : Andy
Student ID    : 1106374
Course        : Diploma in Business Administration
Tuition fees : RM 11250.00
Discount      : RM 0.00
Total fees    : RM 11250.00
```

Continued.....

QUESTION 2 (25 Marks)

Write a **complete C program** that assists the user to determine a simple budget based on the income entered. Using the system, the income would be divided into four different categories;

- Necessity: Typical expenses such as rent, transport, utilities, insurance, etc.
- Invest: Investment opportunity in real estate, stock market, etc.
- Play: Amount that can be spent on small luxury such as fine dining, gadget, etc.
- Charity: Contribution to help other less fortunate person or charitable organization.

Guideline.

- Create a structure called *Budget*. The variables in the structure are:
  - *name* : string size 20
  - *age* : int
  - *income* : float
  - *necessity* : float
  - *play* : float
  - *invest* :float
  - *charity* : float
- Create a structure variable array called *info* with 3 elements in the structure itself.

- In *main()* function:
  - Declare all necessary variables.
  - Open an existing file called *Report.txt* for updating.
    The file contains the name, age, income, necessity, play, invest, charity.

Content of Report.txt before program begins						
Jimmy	25	2750.00	1787.50	550.00	275.00	137.50
Johan	29	3500.00	2275.00	700.00	350.00	175.00

  - If the file can't be opened, display error message "*File cannot be opened. Terminating...*" and quit the program.
  - Call function *get_details(...)* and pass array *info* as argument.
  - Using a *for..loop* (repeat for 3 times):
    - Write the *name, age, income, necessity, invest, play* and *charity* into file *report.txt*.
  - Call function *display(...)* and pass array *info* as argument.
  - Close file *Report.txt*.

- In the *get_details(...)* function:
  - Using a *while..loop (*repeat for 3 times*)*:
    - Request from the user to enter data for *name, age* and *income* for each individual.

Continued.....

- Calculate the allocation amount according to the following table.

Category	Allocation
necessity	65% of income
invest	20% of income
play	10% of income
charity	5% of income

- In the *display(...)* function:
  - Using a *do-while loop (*repeat for 3 times*)*:
    - Display allocated amount for *necessity, invest, play* and *charity* for each individuals.

OUTPUT SCREEN

```
==========================================
     Automated Money Management System
==========================================

Enter your name                  : Mariam
Enter your age                   : 33
Enter your monthly net income    : RM 4000

Enter your name                  : Mindy
Enter your age                   : 37
Enter your monthly net income    : RM 5600

Enter your name                  : Mark
Enter your age                   : 41
Enter your monthly net income    : RM 6250

Mariam, your salary should be allocated as follows:
[1] Necessity allocation  : RM 2600.00
[2] Invest allocation     : RM 800.00
[3] Play allocation       : RM 400.00
[4] Charity allocation    : RM 200.00

Mindy, your salary should be allocated as follows:
[1] Necessity allocation  : RM 3640.00
[2] Invest allocation     : RM 1120.00
[3] Play allocation       : RM 560.00
[4] Charity allocation    : RM 280.00

Mark, your salary should be allocated as follows:
[1] Necessity allocation  : RM 4062.50
[2] Invest allocation     : RM 1250.00
[3] Play allocation       : RM 625.00
[4] Charity allocation    : RM 312.50
```

Continued.....

Content of Report.txt after program terminates						
Jimmy	25	2750.00	1787.50	550.00	275.00	137.50
Johan	29	3500.00	2275.00	700.00	350.00	175.00
Mariam	33	4000.00	2600.00	800.00	400.00	200.00
Mindy	37	5600.00	3640.00	1120.00	560.00	280.00
Mark	41	6250.00	4062.50	1250.00	625.00	312.50

End of Page.